



Guía Docente				
Datos Identificativos				2014/15
Asignatura (*)	Concorrenca e Paralelismo		Código	614G01018
Titulación				
Descritores				
Ciclo	Período	Curso	Tipo	Créditos
Grao	2º cuadrimestre	Segundo	Obrigatoria	6
Idioma	CastelánGalegoInglés			
Prerrequisitos				
Departamento	ComputaciónElectrónica e Sistemas			
Coordinación	Rodríguez Álvarez, Gabriel	Correo electrónico	gabriel.rodriguez@udc.es	
Profesorado	Barreira Rodriguez, Noelia Fraguela Rodriguez, Basilio Bernardo López Taboada, Guillermo Martin Santamaria, Maria Jose Paris Fernandez, Javier Quintela Carreira, Juan Jose Ramos Garea, Sabela Rey Expósito, Roberto Rodríguez Álvarez, Gabriel Tourinho Dominguez, Juan	Correo electrónico	noelia.barreira@udc.es basilio.fraguela@udc.es guillermo.lopez.taboada@udc.es maria.martin.santamaria@udc.es javier.paris@udc.es juan.quintela.carreira@udc.es sabela.ramos.garea@udc.es roberto.rey.exposito@udc.es gabriel.rodriguez@udc.es juan.tourino@udc.es	
Web	campusvirtual.udc.es/moodle			
Descrición xeral	Programación concorrente e paralela. Problemas típicos: interbloqueo, inanición. Semáforos. Comunicación e sincronización. Dependencias. Paso de mensaxes. Algoritmos concorrentes e paralelos.			

Competencias da titulación	
Código	Competencias da titulación

Resultados da aprendizaxe			
Competencias de materia (Resultados de aprendizaxe)			Competencias da titulación
Coñecemento e aplicación dos procedementos algorítmicos básicos das tecnoloxías informáticas para deseñar solucións a problemas, analizando a idoneidade e a complexidade dos algoritmos propostos dende o punto de vista da concorrentia e o paralelismo.	A12		
Coñecemento e aplicación dos principios fundamentais e técnicas básicas da programación paralela, concorrente, distribuída e de tempo real.	A20		

Contidos	
Temas	Subtemas



T1. Principios de programación concurrente	<ul style="list-style-type: none">1.1 Conceptos.<ul style="list-style-type: none">1.1.1 Arquitecturas hardware1.1.2 Sistemas operativos1.1.3 Procesos e Threads1.2 Programación Multiproceso (fork/join)1.3 Programación Multithread1.4 Sección crítica.1.5 Exclusión mutua1.6 Instruccións atómicas1.7 Sincronización mediante condicións1.8 Semáforos<ul style="list-style-type: none">1.8.1 Mutex1.8.2 Semáforos1.9 Interbloqueo. Prevención, evitación, recuperación.1.10 Inanición.1.11 Comunicación e sincronización1.12 Escalabilidade
T2. Algoritmos concurrentes	<ul style="list-style-type: none">2.1 Productores/consumidores.2.2 Lectores/escritores2.3 Filósofos ceando2.4 Compartición nula
T3. Principios de programación paralela	<ul style="list-style-type: none">3.1 Conceptos<ul style="list-style-type: none">3.1.1 Niveis de paralelismo3.1.2 Dependencias de datos3.2 Modelo de paso de mensaxes<ul style="list-style-type: none">3.2.1 Conceptos básicos3.2.2 Comunicación punto a punto3.2.3 Operacións colectivas3.3 Análise de algoritmos paralelos<ul style="list-style-type: none">3.3.1 Medidas de prestacións de algoritmos paralelos3.4 Metodoloxía da programación paralela<ul style="list-style-type: none">3.4.1 Descomposición en tarefas3.4.2 Asignación de tarefas3.4.3 Técnicas de optimización3.5 Esquemas algorítmicos paralelos<ul style="list-style-type: none">3.5.1 Pool de tarefas3.5.2 Paradigma Master/slave3.5.3 Esquemas en árbore3.5.4 Single Process Multiple Data
T4. Deseño de algoritmos e aplicacións paralelas	<ul style="list-style-type: none">4.1 Bibliotecas de paso de mensaxes4.2 Casos de estudo4.3 Evaluación de rendemento4.4 Aplicación de técnicas de optimización

Planificación			
Metodoloxías / probas	Horas presenciais	Horas non presenciais / traballo autónomo	Horas totais
Sesión maxistral	25	40	65
Proba mixta	3	0	3



Prácticas de laboratorio	16	24	40
Solución de problemas	10	20	30
Seminario	5	6	11
Atención personalizada	1	0	1

*Os datos que aparecen na táboa de planificación son de carácter orientativo, considerando a heteroxeneidade do alumnado

Metodoloxías	
Metodoloxías	Descrición
Sesión maxistral	Exposición oral complementada co uso de medios audiovisuais e a introdución de algunhas preguntas dirixidas aos estudantes, coa finalidade de transmitir coñecementos e facilitar a aprendizaxe.
Proba mixta	Exame da materia que combinará preguntas sobre a teoría con problemas a resolver.
Prácticas de laboratorio	Metodoloxía que permite que os estudantes aprendan efectivamente a través da realización de actividades de carácter práctico, tales como desenvolvemento de exercicios de programación.
Solución de problemas	Resolución de problemáticas concretas xurdidas nas prácticas de laboratorio a teor dos coñecementos que se traballaron, os cales poden plantexar múltiples solucións.
Seminario	Presentación en grupo dun tema para o seu estudo intensivo, buscándose a discusión, e a participación dos asistentes ao seminario.

Atención personalizada	
Metodoloxías	Descrición
Prácticas de laboratorio Seminario Solución de problemas	Nas prácticas de laboratorio, seminarios e na resolución de problemas os estudantes poderán presentar cuestións, dúbidas, etc. O profesor/a, atendendo a estas solicitudes, repasará conceptos, resolverá novos problemas ou utilizará calquera actividade que considere axeitada para resolver as cuestións plantexadas.

Avaliación		
Metodoloxías	Descrición	Cualificación
Prácticas de laboratorio	<p>Realización de varios exercicios prácticos de concorrencia (bloque de prácticas de concorrencia) e unha serie de exercicios prácticos de paralelismo (bloque de prácticas de paralelismo). Cada bloque de prácticas representa a metade da nota máxima de prácticas de laboratorio.</p> <p>Inda que a realización das prácticas pode ser por parellas a súa avaliación é individual.</p> <p>É condición necesaria (pero non suficiente) para aprobar a asignatura obter como mínimo o 50% da puntuación máxima de cada un dos dous bloques de prácticas de laboratorio.</p> <p>En caso de NON SUPERAR calquera bloque de prácticas non será posible superar a materia en xuño. En caso de que o alumno se presente ao exame de maio a súa calificación en actas será SUSPENSO. Se non se presenta a súa calificación será NON PRESENTADO.</p>	30
Proba mixta	<p>Proba sobre os contidos presentados na docencia expositiva e desenvolvidos nas sesións de laboratorio e nas titorías grupais. Estará dividida en dous bloques: concorrencia (temas 1 e 2) e paralelismo (temas 3 e 4), sendo cada bloque a metade da nota máxima da proba mixta.</p> <p>É condición necesaria (pero non suficiente) para aprobar obter como mínimo o 30% da puntuación máxima de cada un dos dous bloques, así como obter un mínimo do 40% da puntuación máxima da proba mixta.</p>	70

Observacións avaliación



É condición necesaria (pero non suficiente) para aprobar obter unha cualificación global, aplicando as ponderacións da proba obxectiva e das prácticas de laboratorio, igual ou superior ao 50%. En caso de obter unha cualificación global igual ou superior ao 50% pero non cumprir con algunha das condicións necesarias a cualificación final será un Suspenso (4.5).

O mecanismo de recuperación que se habilita é exclusivamente a repetición da proba mixta (70% da cualificación global da asignatura), requeriéndose de novo como mínimo o 30% da puntuación máxima de cada un dos dous bloques, así como obter un mínimo do 40% da puntuación máxima da proba mixta. En caso de ter obtido como mínimo o 50% da nota dun bloque no exame de maio conservarase esa nota se o alumno non contesta a ese bloque no exame de recuperación de xullo.

A cualificación de prácticas de laboratorio (30% da cualificación global da asignatura) obtida na convocatoria ordinaria consérvase para a recuperación. Non é posible recuperar/mellorar para xullo a nota de prácticas de laboratorio xa que é unha nota obtida en avaliación continua.

Fontes de información

Bibliografía básica	<ul style="list-style-type: none">- Francisco Almeida [et al.] (2008). Introducción a la Programación Paralela. Madrid: Paraninfo Cengage Learning- Peter S. Pacheco (1997). Parallel Programming with MPI. San Francisco, California : Morgan Kaufman- Doug Lea (2001). Programación Concurrente en Java: Principios de Diseño y Patrones. Madrid: Addison Wesley- Joe Armstrong (2007). Programming Erlang: Software for a Concurrent World. United States: Pragmatic Programmers
Bibliografía complementaria	<ul style="list-style-type: none">- Wilkinson, B. y Allen, M.. (1999). Parallel Programming. Techniques and Applications Using Networked Workstations and Parallel Computers. . Upper Saddle River, New Jersey : Prentice Hall,

Recomendacións

Materias que se recomenda ter cursado previamente

Internet e sistemas distribuídos/614G01023

Materias que se recomenda cursar simultaneamente

Sistemas Operativos/614G01016

Redes/614G01017

Proceso Software/614G01019

Materias que continúan o temario

Programación II/614G01006

Algoritmos/614G01011

Estrutura de Computadores/614G01012

Paradigmas de Programación/614G01014

Deseño Software/614G01015

Observacións

(*A Guía docente é o documento onde se visualiza a proposta académica da UDC. Este documento é público e non se pode modificar, salvo casos excepcionais baixo a revisión do órgano competente dacordo coa normativa vixente que establece o proceso de elaboración de guías